409-CD-610-002

EOSDIS Core System Project

ECS Science Acceptance Test Plan for Release 6B

September 2000

Raytheon Company
Upper Marlboro, Maryland

ECS Overall Acceptance Test Plan for Release 6B

September 2000

Prepared Under Contract NAS5-60000
CDRL Item #069

RESPONSIBLE ENGINEER

John Russey /s/ 9/12/00

John Russey, Systems Engineering Department Date
EOSDIS Core System Project

SUBMITTED BY

Mark McBride /s/ 9/12/00

Mark McBride, Director, Systems Engineering Date
EOSDIS Core System Project

Raytheon Company
Upper Marlboro, Maryland

This page intentionally left blank.

Preface

This document is a formal contract deliverable with an approval code 1. It requires Government review and approval prior to final contract acceptance. This document is under ECS contractor configuration control. Contractor approved changes are handled in accordance with the change control requirements described in the EOS Configuration Management Plan. Changes to this document will be made by document change notice (DCN) or by complete revision.

Any questions should be addressed to:

Data Management Office
The ECS Project Office
Raytheon Systems Company
1616 McCormick Drive
Upper Marlboro, Maryland 20774-5301

This page intentionally left blank.

Abstract

The Release 6B ECS System Acceptance Test Plan (ATP) describes the approach that the Test Engineering (TE) Organization takes to verify applicable ECS Release 6B Acceptance Criteria. This Plan contains the overall acceptance test plan, processes, test cases and schedules that will be used to verify Release 6B functionality. The ATP specifies the method used to accomplish the Acceptance Testing of Release 6B. It defines the plan that will be used to formally verify that Release 6B meets the specified operational, functional, and interface requirements.

Keywords: AM-1, Landsat 7, scenario, sequence, test case, acceptance, management, plan, verification, Release 6B, ATP, AT, TE.

This page intentionally left blank.

Change Information Page

List of Effective Pages	
Page Number	**Issue**
Title	Original
iii through xii	Original
1-1 and 1-2	Original
2-1 through 2-4	Original
3-1 through 3-8	Original
4-1 through 4-8	Original
5-1 through 5-6	Original
A-1 through A-16	Original
B-1 through B-4	Original

Document History			
Document Number	**Status/Issue**	**Publication Date**	**CCR Number**
409-CD-610-001	Original	August 2000	00-0911
409-CD-610-002	Update	September 2000	00-0911

This page intentionally left blank.

Contents

4. Test Tools

5. Test Preparation and Execution

List of Figures

List of Tables

Appendix A. Acceptance Test Summaries

Appendix B. Primavera Schedule Listing

This page intentionally left blank.

1. Introduction

1.1 Identification

This Acceptance Test Plan (ATP), Contract Data Requirement List (CDRL) item 069, whose requirements are specified in Data Item Description (DID) 409/VE1, is a required deliverable under the Earth Observing System Data and Information System (EOSDIS) Core System (ECS) Contract NAS5-60000. The Release 6B ECS Science Acceptance Test Plan describes the approach Test Engineering (TE) will take to verify applicable 6B Acceptance Criteria. The Release 6B ECS System Acceptance Test Plan contains the overall acceptance test plan, processes, test cases and schedules used to verify Release 6B.

1.2 Scope

The Release 6B system provides additional capabilities above those provided in the Release 6A system. New major capabilities provided by Release 6B are delineated in paragraph 3.1.2. In addition to the new capabilities introduced in Release 6B, the system will include modifications to address certain NCRs that have been written during prior releases against the ECS system.

1.3 Purpose

The purpose of this Release 6B ECS System Acceptance Test Plan is to provide an overview of the overall acceptance test philosophy, process and schedule used to formally verify that the ECS Release 6B satisfies all criteria based on requirements as delineated in the 6B Science System Release Plan for the ECS Project, 334-CD-610-002.

1.4 Status and Schedule

The submittal of DID 409/VE1 meets the milestone specified in the Contract Data Requirements List (CDRL) for ECS Overall System Acceptance Test Plan of NASA contract NAS5-60000. The submittal schedule is in accordance with the 6B Science System Release Plan, and the Master Program Schedule, as shown in Appendix B, the Primavera Schedule Listing.

1.5 Organization

The Release 6B ECS System Acceptance Test Plan is organized in five sections and two appendices. Sections 1-5 address the approach the TE takes to test the Release 6B ECS system. These sections apply to testing at all locations and include the following detail:

- Section 1. Introduction -- Provides information regarding the identification, scope, purpose, status and schedule, and organization of this document.
- Section 2. Related Documentation -- Provides a listing of parent documents, applicable documents, and documents which are used as source information.

- Section 3. Acceptance Test Overview -- Describes Release 6B capabilities and provides an overview of the acceptance tests.
- Section 4. Test Tools -- Describes the test tools used by TE to conduct ECS Release 6B System Acceptance Tests.
- Section 5. Test Preparation and Coordination -- Discusses the process by which formal acceptance testing is managed and executed.
- Appendix A; Acceptance Test Summaries – Provides a mapping of the Release 6B Test Cases to their associated Acceptance Criteria.
- Appendix B; Primavera Schedule Listing – Provides a print out of the Release 6B Acceptance Test schedule current at the date of print of this document.

2. Related Documentation

2.1 Parent Documents

The parent documents are the documents from which the scope and content of this document are derived.

334-CD-610	6B Science System Release Plan for the ECS Project
194-401-VE1	Verification Plan for the ECS Project, Final
420-05-03	Earth Observing System (EOS) Performance Assurance Requirements for EOSDIS Core System (ECS)
423-41-01	Goddard Space Flight Center, EOSDIS Core System (ECS) Statement of Work
423-41-02	Goddard Space Flight Center, Functional and Performance Requirements Specification for the Earth Observing System Data and Information System (EOSDIS) Core System (ECS)
423-41-03	Goddard Space Flight Center, EOSDIS Core System Contract Data Requirements Document

2.2 Applicable Documents

The following documents are referenced within this Test Procedures document, or are directly applicable, or contain policies or other directive matters that are binding upon the content of this document.

313-CD-610	Release 6B ECS Internal Interface Control Document for the ECS Project, Draft
607-CD-001	Maintenance and Operations Position Description for the ECS Project
211-TP-005	Transition Plan 4PX to 4PY, 4PY to 5A, and 5A to 5B for the ECS Project
http://ecsv.gsfc.nasa.gov/ecsv_v2/rports/db_tickets/index.cgi	5B, 6A and 6B Tickets
http://ecsv.gsfc.nasa.gov/ecsv/index.html	ECS Verification Database (VDB 1)
http://ecsv.gsfc.nasa.gov/ecsv_v2/index.html	ECS Verification Database (VDB 2)
http://dmserver.gsfc.nasa.gov/relb_it/6b.html	Release 6B Acceptance Test Procedures
http://dmserver.gsfc.nasa.gov/ecstest/	ECS Test Data Home Page
CM-1-016-1	"Software Development Using ClearCase" ECS Work Instruction

SD-1-030	“Software Turnover Process” ECS Project Instruction
TT-1-001	“Acceptance Test Preparation, Execution, and Documentation” ECS Project Instruction
TT-1-001-1	"Creating and Updating Acceptance Test Cases" ECS Work Instruction
TT-1-001-2	“Acceptance Test Execution Activities” ECS Work Instruction
TT-1-001-3	“Criteria Verification Log” ECS Work Instruction
TT-1-003-1	“Test Folders” ECS Work Instruction
505-41-11	Goddard Space Flight Center, Interface Requirements Document Between Earth Observing System Data and Information System (EOSDIS) and Version 0 System
505-41-12	Goddard Space Flight Center, Interface Requirements Document Between EOSDIS Core System (ECS) and Science Computing Facilities
505-41-13	Goddard Space Flight Center, Interface Requirements Document Between the Earth Observing System Data and Information System (EOSDIS) and the Landsat 7 System
505-41-18	Goddard Space Flight Center, Interface Requirements Document Between Earth Observing System Data and Information System (EOSDIS) and MITI ASTER GDS Project
505-41-19	Goddard Space Flight Center, Interface Requirements Document Between the EOSDIS Core System (ECS) and the National Oceanic and Atmospheric Administration (NOAA) Affiliated Data Center (ADC)
505-41-21	Goddard Space Flight Center, Interface Requirements Document Between EOSDIS Core System (ECS) and NASA Institutional Support Systems (NISS)
505-41-22	Goddard Space Flight Center, Interface Requirements Document Between the EOSDIS Core System (ECS) and the Stratospheric Aerosol and Gas Experiment (SAGE III)
505-41-30	Goddard Space Flight Center, Interface Control Document Between the EOSDIS Core System (ECS) and the V0 System for Interoperability
505-41-31	Goddard Space Flight Center, Interface Control Document Between the EOSDIS Core System (ECS) and NSI
505-41-32	Goddard Space Flight Center, Interface Control Document Between Earth Observing System Data and Information System, and the Landsat 7 System
505-41-33	Goddard Space Flight Center, Interface Control Document Between the EOSDIS Core System (ECS) and Science Computing Facilities (SCF)

505-41-34	External Interface Control Document Between EOSDIS Core System (ECS) and ASTER Ground Data System
505-41-36	Goddard Space Flight Center, Interface Control Document Between the EOSDIS Core System (ECS) and the National Oceanic and Atmospheric Administration (NOAA) ADC for the ECS Project
505-41-39	Goddard Space Flight Center, Interface Control Document Between the EOSDIS Core System (ECS) and the Langley Research Center (LaRC) Distributed Active Archive Center (DAAC) for the ECS Project
505-41-40	Goddard Space Flight Center, Interface Control Document Between the EOSDIS Core System (ECS) the Goddard Space Flight Center (GSFC) Distributed Active Archive Center (DAAC) for the ECS Project
505-41-47	Goddard Space Flight Center, Interface Control Document Between the EOSDIS Core System (ECS) and the Stratospheric Aerosol and Gas Experiment (SAGE III) Mission Operations Center (MOC)
510-ICD-EDOS/EGS	Earth Observing System (EOS) Data and Operations System (EDOS) Interface Control Document Between the Earth Observing System (EOS) Data and Operations System (EDOS) and the EOS Ground System (EGS) Elements
552-FDD-96/010R0UD0	Goddard Space Flight Center, Earth Observing System (EOS AM-1 Flight Dynamics Division (FDD)/EOSDIS Core System (ECS) Interface Control Document
560-EDOS-0211.0001	Goddard Space Flight Center, EDOS Interface Requirements Document (IRD) Between the Earth Observing System (EOS) Data and Operations System (EDOS), and the EOS Ground System (EGS) Elements
DAO IRD/ICD	(tbd)
DID515	Availability Models Predictions
DID516	Reliability Predictions
DID518	Maintainability Predictions

2.3 Information Documents

The following documents, although not referenced herein and/or not directly applicable, amplify or clarify the information presented in this document, but are not binding on the content of this ECS System Acceptance Test Plan document.

420-TP-019	The Transition Approach to the ECS Drop 5A System, Technical Paper

This page intentionally left blank.

3. Acceptance Test Overview

3.1 Acceptance Test Overview

3.1.1 System Verification Approach

To verify that the ECS system satisfactorily supports the functions specified by the 6B Level 3 and Level 4 requirements, ECS has incorporated a verification database schema to capture related requirement sets in "Tickets" similar to those generated in Release 5B and 6A. A 'Ticket' represents each set of Level 3's, Level 4's, and IRDs that comprise the release capability features for that set of requirements. The ECS Architect Office generates verification acceptance criteria related to the requirements and incorporates them in each Ticket. ESDIS reviews and approves the acceptance criteria. The test organization identifies Test Cases to verify these acceptance criteria, which in turn, are linked in the Verification Database (VDB). The test verification database schema is depicted in Figure 3.1.1.

The Test Team works with the Development Organization during Development integration period to become familiar with each subsystem and informally witness and assist the integration tests as they are developed and performed. These integration tests have become a vital input and building block for the Test Team in refining their acceptance tests. Once defined, the acceptance test cases are scheduled in Primavera. The Test Procedures are then developed by the Test Team. Following an iteration of reviews by the AO and ESDIS with updates by the Test Team, the procedures are approved by ESDIS and posted on the WEB. Acceptance tests are dry run and then formally executed in the Verification and Acceptance Test Center (VATC), Performance Verification Center (PVC), or applicable DAACs if necessary. The test results are reported in the VDB, presented at the CSR, and recorded in the Acceptance Test Report (DID 412).

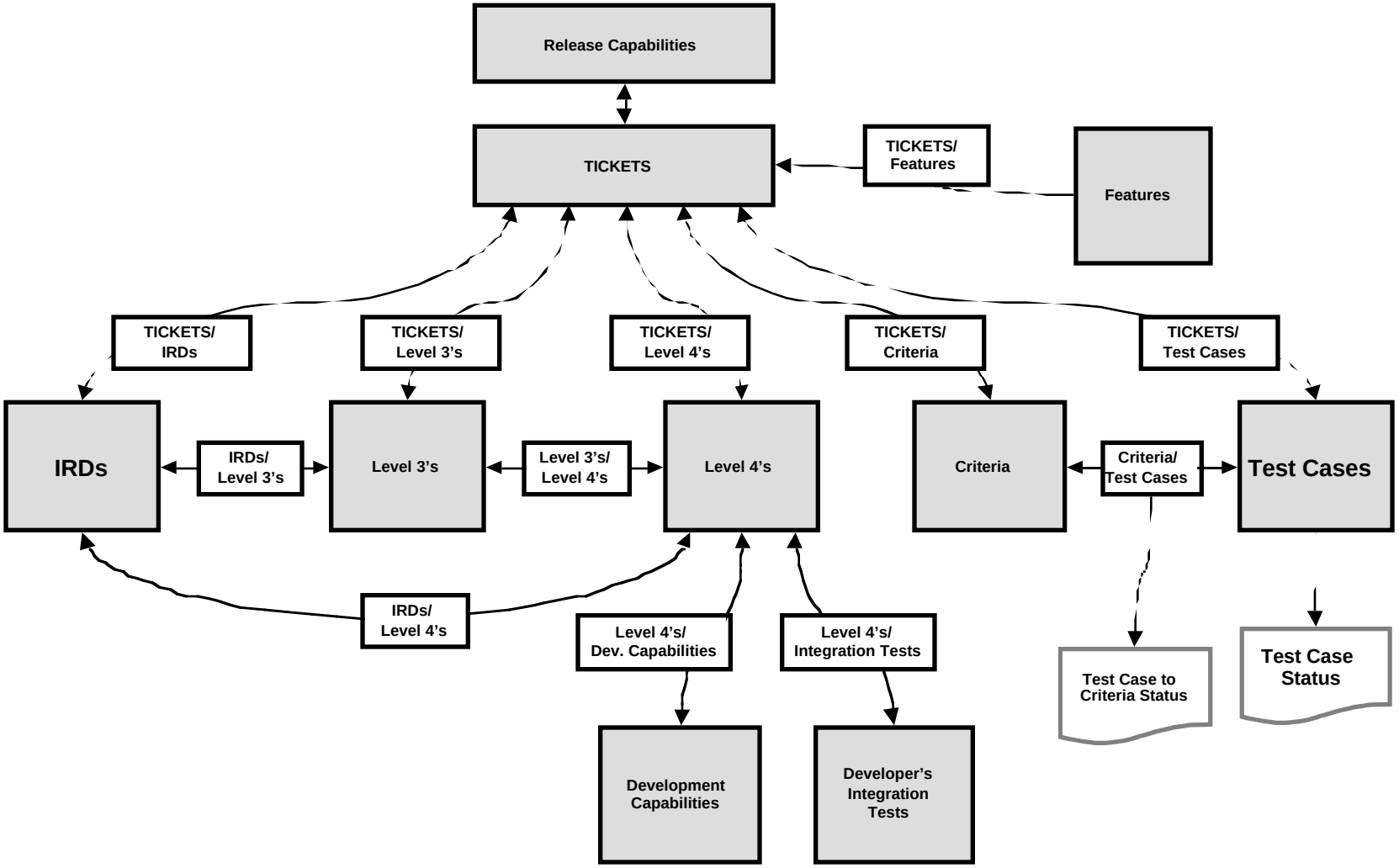


Figure 3.1-1. 6B System Verification Database Schema

3.1.2 Release Capability Priorities

Release 6B is being developed based on a set of Release Capabilities (RC's). These RC's are defined in support of operational readiness for new missions and enhancement of existing capabilities in use by operations. The following provide RC's and their summary description for 6B:

1. **SDSRV Failed Acquire Notification –** Provide notifications to users via email when acquires fail.

2. **Ingest of 6B Data Types** –The SIPS interface will be used to ingest higher-level products produced on SIPS systems, i.e., additional DAS "First-Look" and "Late-Look" products.

3. **Enhanced Fault Recovery-Failover –** Provide the capability to failover from primary to backup machines in the event that there is a failure of a primary machine. (Automated failover recovery capability is pending upon the outcome of ECS's RMA analysis study. Currently, it is unclear whether custom code will be needed to implement this capability.)

4. **Processing 6B System Throughput –** Provide changes to support the increasing requirements of Terra instruments, and Aqua instruments with regard to ingest, production, storage, and distribution, specifically, DPS performance enhancements.

5. **SDSRV Results Set Chunking** – Provide capability to return search results to the V0 Gateway in parts to improve overall throughput and reduce amount of time needed for users to receive search results.

6. **SDSRV Request Priority** – Provide performance enhancements to support the capability for operators to change the priority of a request.

The missions supported by Release 6B are shown in Table 3.1-1. There are no new interfaces to be supported by Release 6B.

Table 3.1-1. Releases 6B Launch Highlights

Satellite	Launch Date	SSI&T	Operations Version	6B Performance Capabilities
Landsat-7	15-Apr-99	N/A	4 or later	Full Ingest & Archive
Terra	18-Dec-99	4 or later	4 or later	Full Ingest & Archive for Processing & Reprocessing
Meteor/ SAGE III	TBD	SIPS I/F Testing	4 or later	Full Ingest & Archive for Processing & Reprocessing
FOO/ ACRIM	20-Dec-99	SIPS I/F Testing	5A or later	Full Ingest & Archive for Processing & Reprocessing
Aqua	31-Dec-00	5B (6A for MODIS)	6A	Processing (1X for L1 and higher) and Reprocessing (1.2X for L1 & 0.65X for higher level)
ICESat GLAS	30-Jul-01	SIPS I/F Testing	6A	Ingest and archive for Processing (1.2X for L0, 1X for L1 and higher) and Reprocessing(1.2X for L1 and 0.65X for higher level)

3.2 Release 6B Requirements

Release 6B has been structured as an upgrade to Release 6A. The Release 6B system provides additional capabilities above those provided in the Release 6A system. These additional capabilities are linked to criteria for Release 6B (as currently planned) and identified in Appendix A. In addition to the new criteria to be verified in Release 6B, the system will have to be tested to verify modifications that addressed certain NCRs that have been written against the ECS system. Future NCR fixes for Release 6B will be identified based upon NCR priority and severity.

3.3 Release 6B Acceptance Test Approach

The objective of the ECS formal testing activity is to verify that the 6B software release is compliant with the established Acceptance Criteria delineated in the Release 6B Requirements Acceptance Tickets ("Tickets").

Systems Engineering (SE) produces the Tickets based on a traceability of the IRDs to Level 3 Requirements, and Level 3 Requirements to Level 4 requirements that will ultimately be baselined in the VDB. SE generates Acceptance Criteria for each Ticket that, in accordance with ESDIS approval, are formulated to verify the requirements associated with that Ticket. The Test Organization is responsible for delineating the test cases that will be generated to test the Acceptance Criteria.

The testing process consists of procedure development, review, updates, ESDIS approval, dry run and formal test. As the Development organization moves into the integration phase of its activities, it begins execution of the integration procedures. By working with Development's integration team, the Test Team uses this information to finalize the acceptance test procedures.

The hardware and software environment is analyzed to determine the expected fidelity of VATC testing and identify any test that, due to the DAAC-specific nature of the acceptance criteria, needs

to be executed in the PVC or one or more DAAC environments. The Test Team uses this analysis, the Acceptance Criteria in the Tickets, and the preliminary test procedure list to delineate the final list of tests to be performed.

By observation and support to Development's integration team effort, the Test Team refines each test procedure by first developing a high level flow of test actions, followed by a functional description of test actions and then by the detailed test actions. The Test Team determines test dependencies and interactions, and sequences. As each test procedure is generated, it is posted to the web, and is the object of an internal peer review and updates before being submitted to the Government for review, comment and approval. The final test procedure is included as part of CDRL 411, ECS System Acceptance Test Procedures. Test cases are identified and summarized in Appendix A and the final procedures are maintained on the web at the following: http://dmserver.gsfc.nasa.gov/relb_it/6b.html

As soon as software is available, the acceptance tests are dry run whenever possible. Following a successful Installation and checkout in the VATC, and when sufficient confidence is attained that the tests can be performed, the test organization schedules the Test Readiness Review (TRR). The TRR is an internal review under the control of ECS systems engineering. The TRR baselines the Government approved revisions/comments to the test procedures. The TRR also establishes the day-by-day sequence of tests to allow for a metrics-based analysis of test program progress. Following a successful TRR, dry runs and formal tests are conducted in the VATC or PVC. All requirements able to be demonstrated in the VATC are addressed as part of formally witnessed tests.

External interface testing is conducted to the extent possible in each test environment under conditions that simulate operational activities. In cases where it is not possible to achieve a reasonable level of fidelity in the VATC, formal verification of acceptance criteria occurs in one or more DAAC environment(s).

With the deployment of the release software to the field, the test organization is on-call to support and assist in the transition activity at each site as needed. Following this on-site activity, a Site Readiness Acceptance (SRA) is conducted to review the completion of the test program at each DAAC. At the SRA, the results of testing are presented. Following the final SRA, the test program concludes with the generation of the ECS Science Acceptance Report, CDRL 412.

3.3.1 ECS Acceptance Criteria Tests

The complete set of ECS Acceptance Criteria allocated to Release 6B is verified to ensure that the release meets those requirements needed to provide specified functionality for the missions addressed in Section 3.1. This includes verifying all test criteria relating to requirements needed to support the ECS Release objectives for data operations, information management and archive, science processing, networks, and system management at each site. Release 6B functionality and error conditions are verified in the VATC using Acceptance Test Procedures.

3.3.2 Transition Verification

The deployment of this release is in accordance with the transition plan, 211-TP-005-005 (or the latest revision in effect). The verification of the 6B release includes the checkout procedures

provided with the software release and DAAC-specified functional verification tests. The latter includes site-specific subsystem checkout procedures, and operational verification procedures.

Transition preparation begins with the design, development, integration, and checkout of the procedures and software (e.g. scripts) required to perform the operational transition of the 6B software release. These activities are the responsibility of Development and are performed initially in the EDF. Concurrently, the Test organization updates the release checkout procedures based on guidance provided by the transition plan. The transition plan identifies functional tests that should be performed after transition, and prioritizes the tests as primary or secondary. Primary tests must be performed after transition whereas secondary tests are performed at the discretion of the DAAC, based on the actual timeline.

Once integrated in the EDF, a second integration and checkout phase begins in the VATC. During this phase, the transition and on-site checkout procedures are exercised and refined. The period in the VATC is also used to train the installation team, including DAAC personnel. This effort results in an improved set of procedures and a cohesive team.

Transition at the DAAC(s) is initiated upon completion of the VATC checkout and the Pre-Shipment Review (PSR). The DAAC staff is responsible for the transition and testing of the release, with support from ECS/Landover. The baseline schedule at each DAAC spans a 3-week period. During the first week, one test mode is transitioned to the new release. The goals during this period include refinement of the transition procedures and additional training for the installation team. The second week includes the transition of the remaining test mode. This transition is used as a dry run of the Ops transition, confirming the capability to perform transition to the target time line. After each of the test mode transitions, the DAAC performs their functional verification tests, including PGE regression testing. Once the test modes have been transitioned to the new release and the DAAC testing is complete, the Ops mode transition is performed. Testing of the Ops transition includes at a minimum the primary checkout procedures. Additional testing, including the secondary checkout procedures and DAAC-specified tests, is performed at the discretion of the DAAC manager.

The local test program concludes with a CSR. The CSR documents the results of the acceptance test program including requirement verification status, liens associated with the release and a lien work-off plan, if needed.

3.3.3 Regression Testing

The purpose of Regression Testing is to exercise the major functions of ECS to provide confidence that the addition of new custom or COTS software does not adversely affect the behavior of unmodified code. The Regression Test Plan provides an overview of the methodology used for the selection, development, and execution of Regression Test activities to be used for 6B.

Regression Test activities are based on normal production scenarios that will exercise ECS functionality. These activities are tailored to each facility and shall contain the following:

- Test Checklist - The purpose of Test Checklists is to provide a list of functional system threads for each subsystem.

- Additional tests that focus on software functions likely to be affected by a new release/update
- Tests focusing on software components that have changed

The Regression Test activities are based on the functional threads found in the checklist. These threads will compose a scenario beginning with ingest, archive, and production, and ending with search, order, and distribution. This scenario will be designed to test the basic functionality of the system after a release or patch is installed. By running this test each time, expected results form a baseline for future regression testing of the system. In addition to the Insertion-Production-Retrieval scenario, several other threads will be developed based on related functions not tested in the core scenario. These threads will be tested only if the new functionality may affect it.

Finally, new functions that are delivered with each new drop or patch will be analyzed, and a determination will be made as to which components could be affected by the new software. Existing regression test cases will be updated to include the new functionality.

Regression testing will be performed after each new software release. Regression testing will also be performed at the DAACs after installation and checkout of 6B after CSR. These regression tests will be tailored to include test cases that exercise specific capabilities of interest to the DAAC, in addition to the general capabilities of the 6B software.

Further details of the Regression testing activity can be found in the Regression Test Plan.

3.3.4 Site Testing

The ECS System Engineering Transition Team will coordinate with each DAAC to plan the on-site delivery of 6B software, including ECS/Landover support for installation and checkout and the transition in operations from 6A to 6B. Deployment of Release 6B is performed in accordance with the 6B Transition Plan.

The ECS Test Team is on call to support and assist the Transition Team wherever needed. The DAAC staff will install the release, then perform integration and checkout in a test mode with the advice and support of the ECS Transition Team staff. Subsequent site testing is the responsibility of the DAAC personnel and includes regression testing tailored for the particular DAAC, with DAAC-specific scenarios.

This testing is generally not an extension of acceptance testing, however, there may be cases where specific acceptance criteria and interfaces must be tested at the DAAC because environmental resources at the Landover test facility could not support the testing.

The details of the on-site activities, such as testing the upgrade of shared modes and transition and rollback testing can be found in the Transition Plan.

3.3.5 COTS Testing

Following the procurement of COTS upgrades is the introduction of the upgrade into Development's domain for analysis, installation, and test within the IDG Cell and the Functionality

Lab. Upon Development organization satisfaction, the product is ready for transition to System Test within the VATC. System Test selects the appropriate tests, and the installation is coordinated with the Infrastructure organization. The COTS testing is not witnessed but a Test Execution Form is completed to document the results. Satisfactory completion of the VATC activities results in the product being prepared for a Pre-Ship Review (PSR). The PSR verifies that all testing and performance milestones have been met and installation instructions prepared and checked out before the product is released for delivery to the customer. A release CCR is generated to accomplish this release. ECS PI CM-1-005 describes the procedures for turnover and installation of COTS products.

Further details of the COTS upgrade activities can be found in the Science System Release Plan.

3.3.6 End to End Testing

ECS plans to perform on-site End to End (ETE) testing for those sites where no testing has been performed earlier. Since there are no new DAACs with Release 6B, ECS has no plans for any DAAC Release 6B ETE testing.

3.3.7 Interface Testing

When required, early interface testing activities take place before formal Acceptance Test. The testing provides an early assessment of the interface compatibility and/or the functionality of the point-to-point systems. Early interface testing permits functionality to be subjected to regression testing, thus eliminating last minute surprises when the formal factory verification test is performed. These integration and test activities are intentionally overlapped with development. There are no interface tests scheduled for 6B.

3.3.8 Performance Testing

Release 6B performance verification will be performed under the purview of the System Engineering Performance Verification Center (PVC) organization and does not fall under the scope of Acceptance Testing. This, however, does not preclude formal verification of acceptance criteria in the PVC. Further details covering Performance Testing can be found in the PVC Plan and the Science System Release Plan (SSRP).

4. Test Tools

4.1 Test Tools

This section identifies and describes the test tools (COTS and custom coded software) used in the execution of the Release 6B ECS Acceptance Tests and the generation of test data sets. The tools for requirements traceability, computer aided software test, configuration management, network status and monitoring, and external interface simulators are discussed below. Table 4-1 summarizes the test tool suite available for Release 6B acceptance tests.

Table 4-1. Release 6B Test Tool Descriptions (1 of 2)

Category	Tool Type	Tool	Tool Description
COTS	Requirements, Capabilities, Criteria, and Test Case Cross Reference Tool	VDB (ECS System Verification Data Base)	The ECS System Verification Database tool provides an audit trail of requirements, capabilities, criteria and test cases to which they are linked. The tool also provides the verification status of all of the above.
COTS	Scheduling and Status Reporting	Primavera	Primavera is used to establish the basic day-by-day testing schedule and the status of acceptance test progress.
COTS	Configuration Management Tool	Clear Case	Clear Case is the principal configuration management tool that uses Version Object Base (VOB) to store the software versions.
COTS	Nonconformance Reporting and Corrective Action Tool	DDTS	DDTS is a UNIX change management and bug tracking system that tracks and manages changes throughout the life cycle of a hardware or software product from initial requirements planning to obsolescence in the field. DDTS works in conjunction with ClearCase.
COTS	Network Management Framework	Whazzup	Whazzup is used to monitor any device that supports the Simple Network Management Protocol (SNMP). This tool aids determining the status of the network and the devices on the network.
COTS	Network Analyzer/Monitor	Network Analyzer/ Sniffer	The Sniffer/Network Analyzer assists in performance testing and monitors and generates traffic on Ethernet and FDDI networks.
COTS	Network Performance Tool	Netperf	Netperf is a benchmark tool that measures various aspects of network performance; primarily focusing on bulk data transfer and request and request/response performance using either the TCP or UDP and the Berkeley Sockets interface.

Table 4-1. Release 6B Test Tool Descriptions (2 of 2)

Category	Tool Type	Tool	Tool Description
COTS	Capture and Playback Automated Test Tool	XRunner	XRunner is an automated software testing system for Xwindow applications. Xrunner automates the full range of software testing needs. Some of the gained functionality includes output synchronization, text recognition, and a high-level testing mode that operates directly on GUI objects.
COTS	Automated Client/Server Testing System	Load Runner	LoadRunner is an automated testing system for client/server applications on UNIX/X platforms. By running multiple users in parallel off the server, LoadRunner enables the automation of load testing, performance testing, and system tuning.
COTS (ETS)	Production Data Set Generation System	SCTGEN	SCTGEN simulates L0 processing systems like EDOS by generating Production Data Sets (PDS).
COTS (SDPF Toolkit)	Level 0 Simulator	L0sim	Generates L0 products in packet format with associated headers.
COTS (HDF Command Library)	HDF File View Tool	vshow	Allows a user to view an HDF file. The output of vshow can be redirected to a file and be printed.
COTS (ECS Development)	HDF File View Tool	EOSview	Allows a user to view an HDF file in selectable parts including metadata portions.
COTS	HDF File View Tool	HDF Browser	Another HDF view tool.
Customized	Metadata Editor & File Generator	mdedit	Allows a user to modify metadata portion of and HDF file and generate any number of additional HDF files.
Customized	Metadata Simulator	(Perl script) and Tcl/Tk	A GUI based tool that allows a user to populate the science data server with realistic metadata.
Customized	Level 0 (packet) View Tool	PDSview, Crview, PKTview	Allows a user to look at packet headers, and construction record(s)
Customized	Level 0 Cloning Tool	Grangen	Allows user to clone many L0 granules from one granule w/PDR
Customized	file dump utility binary file editor	dmp hex	Allows a user to display any file in hexadecimal and ASCII. Allows a user to display and edit any binary file.
COTS	Contains CIL/CAL	XRP	XRP allows to track and audit configuration accountability of ECS hardware and software.
Customized	Installation Tool	ECSAssist	Enables the ECS custom code to be installed.

4.2 Test Planning and Statusing Tools

There are two systems in use on ECS that facilitate the acceptance test planning and statusing process. The use of these tools focus on identifying, cross-referencing, and tracking: capabilities,

requirements, criteria, and test cases. These tools are the ECS System Verification Database (ECSVDB) tool and the Primavera Scheduling tool.

4.2.1 ECS System Verification Database (ECSVDB)

The ECS System Verification Database (ECSVDB) tool provides an audit trail for ECS requirements, capabilities, criteria and test cases. All acceptance test procedures are linked to Acceptance Criteria, which are in turn linked to Level 4 Requirements traceable to Level 3 system requirements via Tickets. ECSVDB contains the official version of all ECS Release 6B requirements and their mapping to test cases. It also contains the status of the ongoing verification process. As test procedures are executed, their impact on the verification status of related criteria are entered into the individual test folders and the Release 6B Criteria Log. The contents of these logs are used to periodically update the requirements verification status in the ECSVDB. A variety of ECSVDB reports can be obtained through contact with the ECSVDB Web site http://ecsv.gsfc.nasa.gov/ecsv-v2/reports/status/index.cgi_ Release 6B acceptance testing will be conducted with primary emphasis on the verification of Acceptance Criteria which consist of functional components, error conditions, and performance constraints, and are linkedto test procedures via the ECSVDB.

4.2.2 Primavera Scheduling Tool

Primavera is the basic scheduling resource used by ECS in scheduling and statusing work progress. All test procedure preparation and conduct is scheduled using Primavera. As testing proceeds, Primavera is up-dated to record progress. Primavera also links capabilities being implemented by the development activity to the test cases that will verify the capability's existence and effectiveness.

4.3 Computer Aided Software Test and Performance Tools

The Mercury XRunner and LoadRunner tools are computer aided software test and performance test tools used to assist in the automation of testing. XRunner is designed to automate the test process by capturing, in a script file, keyboard, mouse input and system under test (SUT) responses, and then playing back those inputs and comparing the results to those stored in an expected results directory. LoadRunner is used to simulate a large number of actual users, in order to measure the response time of a server in a client/server application. Both tools offer sophisticated programming capabilities through a C based language called Test Script Language (TSL) that can be used to drive the system under test much more extensively than would be possible with manual testing. It also offers the virtue of repeating a test sequence with fidelity. The XRunner and LoadRunner tools also provide very reliable playback of user input.

The primary use of the XRunner tool is the automation of functional tests that involve heavy use of graphical user interfaces. Examples of such user interfaces are the Release 6B DAAC or SMC operator screens.

LoadRunner is utilized for all response time testing that involves the Release 6B desktop GUI and during End-to-End tests that involve large numbers of test and operations personnel at multiple sites.

Upon completion of a test script execution, both XRunner and LoadRunner automatically generate test execution reports. LoadRunner generates performance graphs for analysis.

4.4 Configuration Management Tools

The ECS Configuration Management Organization (CMO) is responsible for the management and control of the Software Development Library (SDL) and the baseline configuration management of hardware and software. The Nonconformance Reporting (NCR) system is administered by System Engineering for the NCR Control Board. Two software tools are used to support these efforts: ClearCase and the Distributed Defect Tracking System (DDTS).

4.4.1 ClearCase

The CMO utilizes ClearCase to manage and control the Software Development Library (SDL) which is the central repository for ECS software including test verification items. ClearCase, an automated software tracking tool, manages multiple versions of evolving software components; tracks which versions were used in software builds; performs builds of individual programs or entire releases according to user-defined version specifications; and enforces site-specific development policies. ClearCase scripts are provided by CMO to be used throughout the software development life cycle in order to standardize and automate the tracking of the information in the SDL. The project instruction CM-1-016-1, Software Development Using ClearCase, describes the SDL, the role of ClearCase in the SDL, and the associated ClearCase scripts.

The following test items are stored and baselined by the CMO, via the Software Turnover Process, as they are finalized.

- Verification documents, including test plans, procedures, scripts, and reports
- Test data sets, software and hardware configuration, including test tools
- Unit-tested components, data sets, hardware configuration, and COTS software
- Verified system builds

Since Acceptance Testing of the ECS is conducted within a baselined configured environment, ClearCase is installed at each test site; and CMO electronically deploys the binary files (executable) of ECS software at each test site. In order to maintain the integrity of the test script and test data, CMO deploys TE's test scripts and test data, in the same manner they deploy ECS binary files. This allows the Acceptance Tester, at each test site, to maintain a baseline of changes to the test script and/or test data for the purpose of local configuration or providing a work around to problems.

4.4.2 Distributed Defect Tracking System (DDTS)

The DDTS records nonconformances and reflects the progress of Nonconformance Reports (NCRs) through resolution and captures necessary information to document that progress.

Through the production of management reports, DDTS provides management visibility and metrics to insure that NCRs are being worked in a timely and effective manner. The policies and procedures governing the usage of DDTS on ECS are defined in the Non-Conformance Tracking Project Instruction (PI), SD-1-014.

4.4.3 Baseline Manager (XRP)

The Baseline Manager is used to record and report the design and as-built operational baselines for ECS. It contains the configuration record for baselined items or products. It tracks products by unique identifier, description, location, and model/version, and it identifies the configured articles that are the components of an approved baseline or assembly. It also provides traceability of baselines and products to previous versions and configurations.

The Baseline Manager tool contains the official Configuration Items List and Configuration Articles List used as a basis for the Functional Configuration Audit and Physical Configuration Audit approved by the ECS CSR and SRA for each release.

4.5 Network Status and Monitoring

The three network tools utilized in acceptance tests are the HP OpenView, Sniffer Network Analyzer, and Netperf. Each are described below.

4.5.1 Whazzup

The Whazzup is a network tool that monitors and controls the entire network environment at each ECS site. As a diagnostic tool, it has the capability to isolate faults quickly. The tool allows the user to display a map of the network environment at that local site for the LSM and the maps of all sites at the SMC. These maps are real-time interactive graphical representations which allow the user to detect network problems as they occur without having to update or refresh the display screen, and to diagnose network connectivity. The tool allows the user to create submaps of the map that can be as small as a software component on the system.

4.5.2 Network Analyzer/Sniffer

The Network Analyzer/Sniffer is a fault and performance management tool that analyzes network activity and identifies problems on multitopology and multiprotocol networks.

4.5.3 Netperf

Netperf is a benchmark tool that measures various aspects of network performance. Its primary focus is on bulk data transfer and request/response performance using either the TCP or UDP and the Berkeley Sockets interface.

4.5.4 ECS Assistant

ECS Assistant is an installation tool that enables the ECS custom code to be installed.

4.6 External Interface Simulators

External interface simulators are used during acceptance testing when the real interfacing system is not available. For Release 6B, the simulators used for acceptance testing are described below.

4.6.1 EOSDIS Test System (ETS)

The ETS is primarily designed to support ECS Release 6B and EOS Ground System (EGS) testing. For Release 6B, pending availability, the Low Rate System and the Multimode Portable Simulator is used for EOC testing. In this configuration, ETS provides simulated telemetry data.

4.7 Data Editor, Generators, and View Tools

During acceptance testing, a variety of data editors, generators and viewing tools are used. Each of these tools is described below.

4.7.1 SCTGEN

SCTGEN is a software package with a graphical user interface that simulates CCSDS and non-CCSDS telemetry for both forward and return link data streams. When used as a test tool for EOSDIS it simulates EOS Data and Operations System (EDOS)-generated data products, such as EDOS Data Units (EDU), Expedited Data Sets (EDS), and Production Data Sets (PDS). SCTGEN provides comprehensive error insertion capabilities. SCTGEN generates test data off-line, and does not present a schedule conflict with operational systems. SCTGEN is a portable software system and requires little in terms of resources. The Simulated Consultative Committee for Space Data Systems (CCSDS) Telemetry Generator (SCTGEN) generates telemetry data files and related sets in various formats according to ECS specifications. In conjunction with other ETS components, SCTGEN supports EOS ground system integration, testing, verification, and validation.

4.7.2 Metadata Editor (mdedit)

The Metadata (mdedit) software test tool allows editing of the metadata portion of HDF files and the creation of any number of new files based upon the begin and end times specified. The mdedit tool uses one file as input and produces any number of output files all the same length and basic structure of the input file with the metadata values possibly modified. The original input file is unchanged, and a log file that encapsulates all the important information of the session is recorded.

4.7.3 L0 Granule Generator (Grangen)

The L0 Granule Generator (Grangen) software test tool allows the modification of the packet start and stop time (spacecraft portions of the construction record and packet time stamp in the secondary header of the packet file of the Production Data Set). The Grangen uses two files as input and then produces a new PDS construction record and packet file(s). The original input files are unchanged, and the new PDS files uses the current time as the set creation time imbedded in the file name and PDS_EDS_ID of the construction record. Grangen allows for PDR product delivery records to be made.

4.7.4 L0sim

The L0sim allows the creation of various level 0 data products, such as CCSDS packet files in various formats. EDOS AM-1 is a sample of the various formats supported. The L0sim runs from the UNIX command prompt. It prompts for input, such as file start and stop date, time interval between packets, APID, and the name of the file containing simulated packet data.

4.8 HDF File Display/View tools

HDF file display and viewing tools are explained below.

4.8.1 EOSview

EOSview is a file-viewing tool that examines and verifies the HDF and HDF-EOS data files. EOSview allows the viewing of the HDF files and individual objects and displaying of all metadata fields and data objects. Attributes and annotations can also be viewed.

4.8.2 HDF Browser

The HDF Browser utility enables the examination of HDF file's hierarchy and components. When an HDF file is opened, the HDF Browser displays the hierarchical structure and organization of the file's contents. The capability for viewing each object in the file is also provided.

4.8.3 vshow

The vshow tool is a command-line utility that is executed from the UNIX shell prompt. Vshow lists and displays information about Vdata objects in a HDF file. In addition, the metadata portion of the HDF file is displayed.

4.8.4 Basic File Display/View/Edit Tools

Basic file display capabilities include UNIX commands such as dmp, hexpert, od-x, od-c, diff and sdiff. These commands can be used for file displays, viewing and editing.

4.9 Test Data

A variety of test data is required to exercise the Release 6B system. This test data will be used in conjunction with the simulators described above to simulate the system. Real test data provided by the instrument teams is used whenever possible. In situations where real data is not available, simulated data or similar heritage data is used for testing. The test data is validated and placed under configuration control prior to test execution. The test data required for each test is stipulated in the respective Test Procedure and the schedule for need dates is identified in the Primavera schedule. The test data is provided to the Test organization by the Science Office.

Further information concerning the availability of data sets can be found on the Test Data Home Page: http://dmserver.gsfc.nasa.gov/ecstest/.

This page intentionally left blank.

5. Test Preparation and Execution

This section describes the process by which formal acceptance testing is managed and conducted. The responsibilities of the test manager and test engineers are also described.

5.1 Acceptance Test Preparation

As a fundamental part of the initial test process, ECS System Engineering performs a detailed requirements analysis which includes reviewing and correcting L3 requirement to L4 mappings. The intent is to ensure that the flow-down fully satisfies the Level 3s and IRDs. System Engineering will then define a set of Acceptance Criteria (AC) for the requirements and group them into Tickets. Verification and sell-off of the L4s, L3s and IRDs will be accomplished by virtue of the verification of Acceptance Criteria defined in the Tickets. ACs are functional, error, or performance in nature. As part of the Test Engineering (TE) organization responsibility, a set of test cases will be developed which satisfy the functional and error ACs for a given Ticket. The performance ACs will usually be verified by the System Engineering Performance Verification Center (PVC) organization. The tests planned to be developed in 6B are listed in Table 5.1-1 and summarized in Appendix A. Wherever possible, one acceptance test is created per Ticket. This mapping is captured and tracked in the VDB along with the requirements and Acceptance Criteria previously discussed.

Upon development of requirement groupings in a Ticket, TE assigns resources to each of the requirement groupings. As initial Tickets are developed, the Systems Engineering Architect's Office (AO) will provide them to the responsible TE personnel. The initial development of test procedures starts with the draft ticket of grouping of the ACs. TE refines the test procedures as the Tickets are developed and approved. The focus is on major capabilities, not underlying subsystems. Functional and error criteria may be verified in the same test. TE provides the design summaries of the test procedures in the Acceptance Test Plan, a draft of which is due at IRR. As generation of the procedures concludes, AO and ESDIS review the test procedures developed by TE, and ESDIS approves the final test procedures. To simplify the process of verification, each test procedure will uniquely cover any ACs associated with each requirement grouping. That is, the ACs associated with each grouping of requirements will be uniquely mapped to one test procedure. A test procedure may cover more than one AC in a group.

For Release 6B internal Peer Reviews of Acceptance Tests will be conducted. A walk-through of the test package will be conducted with the responsible Test Engineer, Architect Office representatives, software development representatives, Quality Assurance and operations personnel participating. This should streamline the review process turnaround time with a single review of the material. The updates to the procedure as a result of these reviews will make the test ready for ESDIS review. The final process for ESDIS approval is the same as the process for test procedures developed for Pre-launch Releases.

Table 5.1-1. 6B Test Case Summary

	Test	Ticket	Level-3 Requirements	Level-4 Requirements	Capabilities	Functional Components	Error Conditions	Performance Constraints	VATC	PVC	GSFC	[illegible]	[illegible]	NSIDC
1	6B08010 - Enhanced Fault Recovery-Failover	RM6B02	TBD	TBD	TBD	TBD	TBD	TBD	X	X				
2	6B09020 - 6B Data Types Ingest	RH6B02	3	4	1	4	0	0	X					
3	6B10020 - SDSRV Failed Aquire Notification	EN6B01	1	8	1	16	1	0	X					
4	6B10030 - Request Priority	EN6B02	6	13	1	7	0	0	X					
5	6B10040 - Results Chunking	EN6B03	1	11	1	5	1	0	X					

TE will follow established standards for documenting test procedures. Each test procedure will identify the incremental steps for verifying the Acceptance Criteria for that test.

TE will conduct dry runs followed by formal tests to verify the approved ACs with designated witnesses and using the approved test procedures. Dry runs are scheduled as soon as pre-turnover release software is available in order to exercise capabilities early and to provide feedback to the development organization in the form of NCRs. The acceptance tests will be performed primarily in the VATC, although some testing of Acceptance Criteria might be done in the PVC also.

The daily progress of the test activity is reviewed by the test leads at internal test status meetings held on Monday, Tuesday and Thursday, and at an all test staff review meeting held each Wednesday. In addition, the daily test status is presented at the Daily Status Reviews (DSR) meeting on Monday through Wednesday with an expanded weekly test presentation given each Thursday.

Regression testing is performed following the installation of each software release, as well as any incremental major software build, to ensure no degradation or modification has occurred to the release already tested. The regression tests are comprised of a representative suite of procedures pulled from the Acceptance Tests.

5.2 Test Execution

The test procedures are derived from the acceptance criteria in the tickets and the software integration test activity. The generation of the test procedures begins with the availability of acceptance criteria and continues as the design and informal tests of the Development organization progresses. Table 5.1-1 delineates the summary of the 6B Release allocation of Level 3, Level 4, capabilities and criteria to test cases.

The software release is installed in a dedicated mode(s) in the VATC for formal testing. Following dry runs to ensure successful execution, formal tests are conducted to verify the set of Release 6B criteria reflected in Appendix A of this plan. All applicable criteria are demonstrated in the VATC as part of formally witnessed tests. External interface testing is conducted to the extent possible in each test environment under conditions that simulate operational activities.

Formal test execution concludes with a Consent to Ship Review (CSR). The CSR documents the results of the VATC test program including verification status, liens associated with the release and a lien work-off plan if needed. Successful conduct of the CSR is predicated on satisfactory Functional Configuration Audit (FCA) results and resolution or agreed-to liens on severity 1 and 2 NCRs. The successful CSR milestone marks the ESDIS approval to ship the Release to the field.

Before deployment of the release, ECS ensures close coordination with each DAAC to plan the on-site delivery. This includes on-site ECS/Landover support for test execution and post-test analyses that may be required. On-site deployment is concluded with the conduct of a Site Readiness Assessment (SRA) at each DAAC. Following the SRA, with the release to the sites in the field, the responsibility and control for the system is turned over to the M&O organization. The M&O staff will conduct the CCB for changes and manage NCR fixes and modifications. Successful completion of the SRA is based on satisfactory PCA results and no severity 1 or 2 NCRs against new capabilities as a result of the site testing.

5.3 ECS Test Process

The ECS Test Process is an interconnected activity flow that incorporates Government and Contractor organizations, different venues and defined interdependencies. The process is described in the following paragraphs.

Historically, the Development organization understands a given software release's functionality, and its "look and feel." The Test Organization is familiar with system level requirements and acceptance criteria, as well as the operational usage of the software. In order to create some "synergism", test resources become involved with development earlier in the Release lifecycle. In this way, test resources become more familiar with a given release much earlier, while Development can gain insight concerning the needs of formal testing. As part of the Release lifecycle, Development will develop integration test scenarios consistent with system level capabilities. Test engineers can then support integration of the software supporting these scenarios, but can also refine Acceptance Test Procedures.

At the completion of the software Turnover and the integration and checkout in the VATC, an internal Test Readiness Review (TRR) will be held. This milestone represents the formal turnover of the software release to the TE Organization. This gate will be strictly monitored to ensure that all integration has been successfully completed, all necessary documentation or installation procedures needed in the VATC are available and approved, required test data and test configurations are available, and any other important information is communicated to the TE Organization prior to the start of formal testing.

Test folders are created for each acceptance test procedure and maintained throughout the remaining Release 6B test program. Each test activity is recorded on Test Execution Forms and filed in their individual test folders. During the test process, discrepancies are noted on the Test Execution Form. Discrepancies are then recorded on NCRs, rated by the test engineer according to severity, and filed in DDTS. Test folders are returned to a secure location, under configuration control, after completion of each test session. Each folder may be subjected to a test folder audit to ensure folder completeness and accuracy.

As-executed procedures and workarounds are documented as a result of test dry runs and formal executions. These are recorded on the test execution sheets that are maintained in the test folder for each test procedure. Workarounds to circumvent system deficiencies found during these tests are recorded on the Test Execution Forms in the test folders of the test case that uncovered the deficiency. The workaround is also recorded on the NCR and recorded in DDTS as part of the NCR process.

The preparation of test results begins with the routine recording of test procedure execution results on the Test Execution Form maintained in the Test Folders. Test results are maintained from the working test level and passed upwards for their incorporation in the Test Folders and DDTS. Acceptance Criteria verification results are then entered on the Criteria log and into the VDB using the WI (TT-1-001-3). This information and others will form the basis for the preparation of acceptance test results.

Whenever possible, the formal verification of all requirements will be accomplished locally in the VATC, or the PVC in the case of Performance Constraints that can not be verified in the VATC.

On rare occasions, due to the DAAC-specific nature of the acceptance criteria, it may be necessary to complete the Release 6B Test Program at a given DAAC as part of scheduled Operational Installation Check-out and Test (ICT) activities. These tests will be coordinated in advance with DAAC, ESDIS, and QA personnel to provide for the necessary formal witness and sign-off environment.

5.4 Release 6B Test Procedures

All 6B test procedures as listed below will be executed in the VATC to verify the criteria that have been incorporated in Release 6B. Appendix A contains a summary of the objective of each Test Case and the mapping to the criteria contained in the appropriate Ticket.

	Test Number - Procedure Title
1	6B08010 - Enhanced Fault Recovery-Failover
2	6B09020 - 6B Data Types Ingest
3	6B10020 - SDSRV Failed Aquire Notification
4	6B10030 - Request Priority
5	6B10040 - Results Chunking

NOTE: The Ticket for test Case 6B08010, Enhanced Fault Recovery-Failover, has been put on hold with activity suspended until October, 2000. It is possible that this test case will need to be executed in the PVC.

5.5 Release 6B Test Schedule

Primavera is the basic scheduling resource used by ECS in scheduling all test activities. A Primavera schedule listing includes the Test case number, test case title, test site, and planned date for each test to be executed for Release 6B Acceptance Testing. Appendix B of this document contains the Release 6B Acceptance Test Schedule listing from Primavera as of the date of print of this document. The Primavera schedule tool should be consulted to obtain up-to-date schedule information.

This page intentionally left blank.

Appendix A. Acceptance Test Summaries

A summary of each test and the mapping of Criteria to Test Cases is attached.

1. **6B08010 - Enhanced Fault Recovery-Failover**

Test Procedure No.: 6B08010	**Ticket Version:** TBD	**Date Executed:**	**Test Conductor:**

Title	Enhanced Fault Recovery Failover
Objective	THE TICKET FOR THIS TEST IS ON HOLD. ACTIVITY HAS BEEN SUSPENDED UNTIL OCTOBER 2000.
	Preconditions: .

Criteria Mapping

Ticket ID	Criteria ID	Criteria Key	Criteria Type	Criteria Text	Test Steps
RM_6B_02			FC		

Test Information

Test Input	➢ ➢ ➢
Test Output	➢ ➢

Data Set Name	Path Name	Description
TBD	TBD	TBD

Test Configuration	➢ ➢

Data Set Name	Path Name	Description
TBD	TBD	TBD

Test Configuration	➢ ➢

2. 6B09020 - 6B Data Types Ingest

Test Procedure No.: 6B09020	**Ticket CCR Num: TBD**	**Date Executed:**	**Test Conductor:**

Title	Ingest of 6B Data Types
Objective	This test case verifies the ability to ingest 6B data (native format only) from the Data Assimilation System (DAS) and National Centers for Environmental Predictions via the electronic (Polling with Deliver Record) interface. Product Delivery Records (PDRs) are generated by the DAS and NCEP data providers and placed in the pre-specified polling directories. The Ingest polling server will periodically poll these directories for new PDRs. Ingest will ignore PDRs that have already been ingested. The data identified in the new PDRs that are detected will be pulled via FTP to a staging directory at the ECS DAAC and will be preprocessed and core metadata files will be generated. The data and core metadata files are sent to the Data Server Subsystem (DSS) for insertion. The data files are archived and the core metadata files are used to update the SDSRV Inventory database. The following DAS data types (all in HDF-EOS format) will be ingested and archived: DFLAPCLD, DFLAPMOM DFLAPMST, DFLAPTMP, DFLAPTRP, DFLASCHM, DFLASCLD, DFLASMIS, DFLASMOM, DFLASMST, DFLASTMP, DFLASTRP, DFLNPANL, DFLNPFCT, DFLFPMIS, DFLFXMIS, DFLN_RST, DFLN_ODS, DLLASCHM, DLLASCL, DLLASMIS, DLLASMOM, DLLASMST, DLLASTMP, DLLASTRP, DLLNPANL, DLLNPFCT, DLLFPMIS, DLLFXMIS, DLLN_RST, DLLN_ODS The following NCEP data types will be ingested and archived: MRF – GRIB format SHBU_OBS – BUFR format NCEP_ICE – GRIB format

Criteria Mapping

Ticket ID	Criteria ID	Criteria Key	Criteria Type	Criteria Text	Test Steps
RH_6B_02	10	TBD	FC	Verify that the system can ingest and archive the NCEP 1-Degree Medium Range Forecast (MRF) data in the GRIB format.	TBD
RH_6B_02	20	TBD	FC	Verify that the system can ingest and archive the NCEP Ship/Buoy Observations (SHBU_OBS) data in the BUFR format.	TBD
RH_6B_02	30	TBD	FC	Verify that the system can ingest and archive the NCEP SSM/I Daily Sea Ice Product (NCEP_ICE) data in the GRIB format.	TBD
RH_6B_02	40	TBD	FC	Verify that the system can ingest and archive the following DAS data types: DFLAPCLD, DFLAPMOM DFLAPMST, DFLAPTMP, DFLAPTRP, DFLASCHM, DFLASCLD, DFLASMIS, DFLASMOM, DFLASMST, DFLASTMP, DFLASTRP, DFLNPANL, DFLNPFCT, DFLFPMIS, DFLFXMIS, DFLN_RST, DFLN_ODS, DLLASCHM, DLLASCL, DLLASMIS, DLLASMOM, DLLASMST, DLLASTMP, DLLASTRP, DLLNPANL, DLLNPFCT, DLLFPMIS, DLLFXMIS, DLLN_RST, DLLN_ODS.	TBD

Test Information

Test Input	• 31 PDRs each containing a different granule of one of the DAS data types listed above. • 1 PDR containing 2 granules of NCEP 1-Degree Medium Range Forecast System, Forecast at 00Z (MRF) data • 1 PDR containing 2 granules of NCEP Ship/Buoy Observations (Locations) (SHBU_OBS) data • 1 PDR containing 2 granules of NCEP SSM/I Daily Sea Ice Product (NCEP_ICE) data
Test Output	• Ingest GUI Monitor/Control Display • Archive directory listings • SDSRV Inventory database queries • PAN E-mail messages

Data Set Name	Path Name	Description
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLAPCLD data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLAPMOM data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLAPMST data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLAPTMP data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLAPTRP data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLASCHM data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLASCLD data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLASMIS data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLASMOM data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLASMST data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLASTMP data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLASTRP data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLNPANL data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLNPFCT data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLFPMIS data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLFXMIS data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLN_RST data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DFLN_ODS data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLASCHM data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLASCL data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLASMIS data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLASMOM data with a PDR

TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLASMST data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLASTMP data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLASTRP data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLNPANL data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLNPFCT data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLFPMIS data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLFXMIS data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLN_RST data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/das	1 granule of DLLN_ODS data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/ncep/mrf	2 granules of MRF data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/ncep/shb	2 granules of SHBU_OBS data with a PDR
TBD	t1icg01:/usr/ecs/OPS/CUSTOM/icl/t1icg01/data/ins_data/data/ncep/nice	2 granules of NCEP_ICE data with a PDR
Test Configuration	• 6B baselined code • Servers (EcInGUI, EcInPolling, EcInReqMgr, EcInGran, EcDsStStagingDiskServer, EcDsStIngestFtpServer, EcDsScienceDataServer, EcDsStArchiveServer, EcIoAdServer,) • Hardware (t1icg03, t1acg04, t1drg03, t1acs02, t1acs03, t1ins02) • Netscape E-mail.	

3. 6B10020 - SDSRV Failed Acquire

Test Procedure No.: 6B10020	Ticket Version: TBD	Date Executed:	Test Conductor:

Title	SDSRV Failed Acquire Notification
Objective	This test case verifies the capability for SDSRV to send an e-mail notification to a user upon encountering a failure in an asynchronous acquire request (the failure is encountered before the request gets to the DDIST subsystem). The failed acquire email is triggered by a non-retryable error being encountered during the SDSRV portion of the acquire, and it is retrieved from the MSS User Profile if it is not included in the request. If a user's email address is not specified or cannot be located, a configurable, default email address will be used. This will also be verified during this test. Both Landsat-7 and non-Landsat-7 data will be acquired, and it will be verified that in the case of Landsat-7 LPS data, a separate email message is sent to the DORRAN system (simulated by a different email address setup in the DDIST configuration registry). The case where the user does not have sufficient privileges to acquire the data (restricted granule access), will also be verified in this test. In this case, the user will receive an email saying he had insufficient privileges to order the data at this time. In addition, the case where data granules have been deleted from the archive will also be attempted. In the case of synchronous acquires (those from the PDPS subsystem), if a failure is encountered during the acquire, the normal procedure of sending a failure status code synchronously to the client for each request will be followed. There will be no email sent on a failed acquire to the client in this case. This will also be verified during this test. In addition, the STMGT server will be terminated and it will be verified that the SDSRV attempts to connect to STMGT (a retryable error), instead of sending a failure email. The STMGT server will then be initiated and the request will complete. Asynchronous acquire requests will be sent from the EOS Data Gateway (EDG), the Machine-to-Machine Gateway (MTMGW), the ASTER GDS to ECS Gateway, and from the Data Server Request Driver during this test. No specific data is needed for this test.

Criteria Mapping

Ticket ID	Criteria ID	Criteria Key	Criteria Type	Criteria Text	Test Steps
EN_6B_01	10	1906	FC	Using the V0 EDG or a test driver, perform a Landsat 7 asynchronous product acquire request that requires subsetting. Design the test such that the SDSRV fails with a non-retryable error when performing data subsetting. Verify two e-mail notifications are sent: one to the user, and one to DORRAN (or a simulated address for DORRAN). Verify the notifications provide the reason for failure. Verify an appropriate e-mail preamble is included in the e-mails. Verify the user e-mail address to which the notification is sent matches the one provided in the request.	TBD

EN_6B_01	20	1907	FC	Using the V0 EDG or a test driver, perform a non-Landsat 7 asynchronous product acquire request. Design the test such that the SDSRV fails with a non-retryable error. Verify an e-mail notification is sent to the user. Verify no e-mail is sent to DORRAN (or a simulated address for DORRAN). Verify the notification provides the reason for failure. Verify an appropriate e-mail preamble is included in the e-mail. Verify the user e-mail address to which the notification is sent matches the one provided in the request.	TBD
EN_6B_01	21	1908	FC	Using the MTMGW or a test driver, repeat the test in Criterion ID 20 and verify the results as described.	TBD
EN_6B_01	22	1909	FC	Using the ASTER GDS to ECS Gateway or a test driver, repeat the test in Criterion ID 20 and verify the results as described.	TBD
EN_6B_01	30	1910	FC	Using the V0 EDG or a test driver, perform an asynchronous acquire request for the data granules that do not exist in the archive. Verify an e-mail notification is sent to the user. Verify the notification provides the reason for failure. Verify an appropriate e-mail preamble is included in the e-mail. Verify the e-mail address to which the notification is sent matches the one provided in the request.	TBD
EN_6B_01	31	1911	FC	Using the MTMGW or a test driver, repeat the test in Criterion ID 30 and verify the results as described.	TBD
EN_6B_01	32	1912	FC	Using the ASTER GDS to ECS Gateway or a test driver, repeat the test in Criterion ID 30 and verify the results as described.	TBD
EN_6B_01	40	1913	FC	Using V0 EDG or a test driver, perform an asynchronous product acquire request for the restricted granules for which the user has no access privileges. Also the order request does not contain the user's e-mail address. Verify an e-mail notification is sent to the user. Verify the notification provides the reason for failure. Verify an appropriate e-mail preamble is included in the e-mail. Verify the e-mail address to which the notification is sent is obtained from the MSS user profile database.	TBD
EN_6B_01	41	1914	FC	Using the MTMGW or a test driver, repeat the test in Criterion ID 40 and verify the results as described.	TBD
EN_6B_01	42	1915	FC	Using the ASTER GDS to ECS Gateway or a test driver, repeat the test in Criterion ID 40 and verify the results as described.	TBD
EN_6B_01	50	1916	FC	Using V0 EDG or a test driver, perform an asynchronous product acquire request for the restricted granules for which the user has no access privileges. Also make sure that neither the order request nor the MSS user profile contain the user's e-mail address Verify an e-mail notification is sent to the default e-mail address configured for the SDSRV. Verify the notification provides the reason for failure. Verify an appropriate e-mail preamble is included in the e-mail.	TBD
EN_6B_01	51	1917	FC	Using the MTMGW or a test driver, repeat the test in Criterion ID 50 and verify the results as described.	TBD
EN_6B_01	52	1918	FC	Using the ASTER GDS to ECS Gateway or a test driver, repeat the test in Criterion ID 50 and verify the results as described.	TBD

EN_6B_01	60	1919	FC	Using the V0 EDG or a test driver, perform an asynchronous product acquire request. Shut down the STMGT server, so that the SDSRV is not able to successfully connect to the server. Verify the SDSRV keeps on attempting to connect to the server (since this is a retryable error) and that no failed acquire e-mail notification is sent to the user. Restart the STMGT server. Verify the processing continues and the desired product is acquired and distributed.	TBD
EN_6B_01	61	1920	FC	Using the MTMGW or a test driver, repeat the test in Criterion ID 60 and verify the results as described.	TBD
EN_6B_01	62	1921	FC	Using the ASTER GDS to ECS Gateway or a test driver, repeat the test in Criterion ID 60 and verify the results as described.	TBD
EN_6B_01	70	1922	EC	Using PDPS or a test driver, perform a synchronous product acquire request. Design the test such that the request fails with a non-retryable error. Verify the proper failure code is returned by the SDSRV. Verify no failed acquire e-mail notification is sent to the user.	TBD

Test Information

Test Input	➢ Search and Acquire Requests submitted from the EDG Client and the DSS Request Driver ➢ Acquire Requests submitted from the MTMGW and the ASTER GDS to ECS Gateway ➢ Acquire Requests submitted from the PDPS subsystem ➢ Forced retryable and non-retryable errors affecting the acquire requests
Test Output	➢ Successful Search Requests ➢ Unsuccessful Acquire Requests ➢ Email messages sent to the user when the asynchronous acquire requests fail in SDSRV ➢ Successful Acquire Requests when appropriate

Metadata/Data Set Name	Path Name	Description
N/A	N/A	N/A

Test Configuration	➢ All standard DAAC servers ➢ t1acs03, t1dps01, t1acg01, t1drg01, t1ins01, t1wkg01, t1sps02 ➢ EDG Client, MTMGW Client Scripts, ASTER Client Scripts, DSS Client Driver ➢ ESDTs: L70RWRS, L70R, MOD00, AST_L1A

4. 6B10030 - Request Priority

Test Procedure No.: 6B10030	Ticket Version: TBD	Date Executed:	Test Conductor:

Title	Request Priority
Objective	This procedure test the capability for the SDSRV CI to service all requests (asynchronous and asynchronous) in order of priority. Each service request have 5 associated priority levels assigned to it which are listed in order of relative priority: Express, Very high, High, Normal, Low. All acquire request are dispatched by the SDSRV based on the priority assigned, and within the same priority level on a first-in-first-out basis. There are several types of requests: Inventory search request, Metadata Insert request, Metadata Update request, Metadata Delete request, Non-subsetting product acquire request, Subsetting product acquire requedst, Browse request, and Inspect request. Each request type is scheduled separately in the SDSRV CI and the number of concurrent requests for each service type will be configured during the server start-start time. The SDSRV GUI uses the service request types when testing the number of concurrent requests. Using the SDSRV GUI the operator staff have the capability to change the priority of a request without having to shut down the server. The operator using the SDSRV GUI can temporally block certain service request types, control the work flow by adjusting the number of concurrent requests assigned to each service type and reset the priority of a pending service request in the work queue. Each scenario will be verified in this test case.

Criteria Mapping

Ticket ID	Criteria ID	Criteria Key	Criteria Type	Criteria Text	Test Steps
EN_6B_02	10	1923	FC	Set the maximum number of concurrent requests for inventory each to zero using the SDSRV GUI. Perform three or more inventory search request in rapid sequential order from multiple V0 EDG clients or test drivers, each request with a different priority level. Verify the requests are pending and are not serviced by the SDSRV. Using the SDSRV GUI, reset the maximum number of concurrent requests for inventory search to 1. Verify using the SDSRV GUI that the inventory search request with the highest priority is now serviced by the SDSRV. In the same manner described above, set the maximum number of concurrent requests for inventory search to 2, 3, so on, and verify that the remaining requests are serviced one after another in priority order. Perform similar tests for insert, update, delete, sub-setting acquire, non-subsetting acquire, browse, and inspect requests, and verify using the SDSRV GUI that the request are serviced by the SDSRV in order of the request priority.	TBD

EN_6B_02	20	1924	FC	Perform three or more inventory seach requests in rapid sequential order from multiple V0 EDG clients or test drivers, each request with the same priority level. Verify using the SDSRV GUI that the requests are serviced by the SDSRV in order of the time at which it receives the request. Perform similar tests for insert, update, delete, sub-setting acquire, non-subsetting acquire, browse, and inspect requests, and verify the requests are serviced by the SDSRV in order of the time at which it receives the request.	TBD
EN_6B_02	30	1925	FC	Start up the SDSRV. Using the SDSRV GUI, verify the maximum number of concurrent request allocated for each service type (search, insert, update, delete, non-subsetting acquire, and sub-setting acquire), and the total number of synchronous service requests that can be queued for execution match the corresponding configured values used to start up the server. Using the SDSRV GUI, reset the maximum number of concurrent requests for an inventory search to 3. Perform 5 concurrent inventory searches. Verify the SDSRV processes no more than 3 concurrent inventory search requests at a time. (Note: The numbers 3 and 5 are provided as an example only. Testers can choose different positive numbers as appropriate.) Perform similar test and verifications as described above for the insert, update, delete, sub-setting acquire, non-subsetting acquire, browse, and inspect request.	TBD
EN_6B_02	40	1926	FC	Perform an inventory search request using the V0 EDG client or a test driver. While the search request is being processed by the SDSRV, set the maximum number of concurrent requests for inventory search to zero using the SDSRV GUI. Verify the ongoing search request proceed to the end and is not affected by the setting. Submit a new inventory search request. Verify the new request is now pending and is not serviced by the SDSRV. Using the SDSRV GUI, reset the maximum number of concurrent request for the inventory search to a non-zero positive value. Verify the request that was held in the work queue earlier is now serviced by the SDSRV. Perform similar tests and verifications as described above for the insert, update, delete, sub-setting acquire, non-subsetting acquire, browse, and inspect requests.	TBD
EN_6B_02	50	1927	FC	Submit three or more inventory search requests in rapid sequential order from multiple V0 EDG clients or test drivers, each request with the priority level. Using the SDSRV GUI, change the priority level of each pending request such that the first request submitted has a lower priority than the last request.	TBD

				Verify the requests are serviced by the SDSRV CI in the order of the changed priority level. Perform similar test and verifications as described above for the insert, update, delete, sub-setting acquire, non-subsetting acquire, browse, and inspect requests.	
EN_6B_02	60	1928	FC	Submit concurrent service requests to the SDSRV including a mix of different services. Include three or more service requests with different priority levels for each of the following services: inventory search, update, delete, sub-setting acquires, non-subsetting acquire, browse and inspect requests. Verify the requests are serviced by the SDSRV CI in priority order in each of the service categories.	TBD
EN_6B_02	70	1929	FC	Using the SDSRV GUI, set the maximum number of concurrent inventory search requests to 5, and the total number of synchronous requests that can be queued for execution to 10. Submit 12 inventory search requests with the same priority to the SDSRV in rapid sequential order. Verify the first 5 request are serviced concurrently, the next five requests are queued, and the last 2 requests receive a re-tryable error from the SDSRV. (Note: The numbers 5,10, and 12 are provided as an example only. Testers can choose different positive numbers are appropriate.) Perform similar test and verifications as described above for update, delete, sub-setting acquire, non-subsetting acquire, browse and inspect requests.	TBD
Test Information					

Test Input	• ESDT • Inventory search requests • Metadata insert requests • Metadata update requests • Metadata delete requests • Non-subsetting product acquire requests • Sub-setting product acquire requests • Browse requests • Inspect requests • Asynchronous acquire requests • Synchronous requests • Priority requests (Express, Very High, High, Normal, Low)	
Test Output	• Error and success messages written to the SDSRV GUI and to the log files.	
Data Set Name	**Path Name**	**Description**
ASTER, Landsat 7	Text	Text
Test Configuration	• Standard VATC configuration - all servers running • SDSRV Sever • MTMGW Client • V0 EDG clients or test drivers • PDPS	

5. 6B10040 - Results Chunking

Test Procedure No.: 6B10040	Ticket Version: TBD	Date Executed:	Test Conductor:

Title	Results Chunking
Objective	The purpose of this test is to verify the capability of the SDSRV to return search results in predetermined increments (chunks) to client CIs. Specifically, the V0 Gateway and MTMGW. This capability reduces response time by allowing the client to process results before the full search routine is completed. Searches will use the EDG Client and the SIPS interface. Searches, specifying all metadata attributes, are requested that exceed the SDSRV default chunk size and with a V0 Gateway default that is smaller than the SDSRV default. Correct search results are received. Searches that exceed SDSRV default should return the SDSRV defaulted value and searches that are less than SDSRV default should return the V0 Gateway determined value. A search is conducted where the data is less than one chunk. The search completes successfully and the data is returned in a single chunk. A search is conducted using a SIPS Interface driver that exceeds a single chunk. The search returns successfully in the size defaulted by the SDSRV. Multiple searches are conducted using the EDG Client and the SIPS test driver concurrently. Searches are successful and in the proper chunk size specified by SDSRV default values. Using the EDG Client, specify a search containing a subset only a subset of the metadata attributes, and containing more granules that the SDSRV default size. The V0 Gateway is configured at less than the SDSRV default size. The search completes successfully in chunks of the size determined by the V0 Gateway. Another search is conducted with the V0 Gateway specifying a chunk larger than the SDSRV default. This search returns successfully at the SDSRV default chunk size. A search is initiated, using the EDG Client, that exceeds the SDSRV default chunk size. The SDSRV fails during the search. The search is suspended until the SDSRV is restarted. Repeat the same search. Fail the V0 Gateway connection after the first chunk is received. An error message is returned to the EDG Client.

Criteria Mapping

Ticket ID	Criteria ID	Criteria Key	Criteria Type	Criteria Text	Test Steps

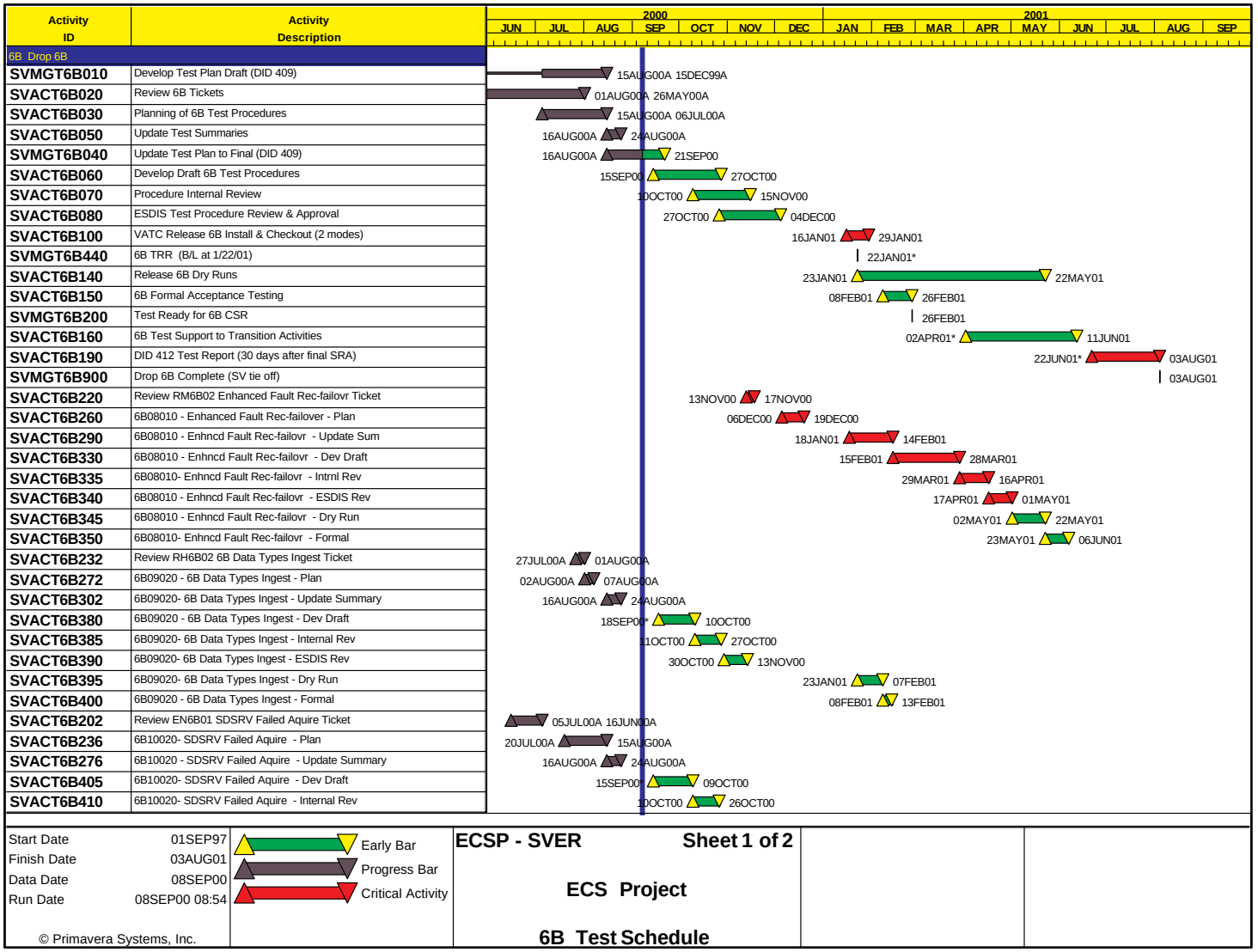
EN_6B_03	10	1887	FC	Using the V0 EDG client or a test driver simulating V0 protocols, perform an inventory search specifying that all metadata attributes are to be returned in the search results. Set up the test case such that the inventory search results contain more granules than the SDSRV default chunk size. Also, let V0 GTWAY pass to the SDSRV a chunk size smaller than the SDSRV default chunk size. Verify correct search results are returned to the V0 EDG client or the test driver. Verify the search results are returned in multiple chunks from the SDSRV to the V0 GTWAY using the chunk size specified by the V0 GTWAY. Repeat the above inventory search test with the V0 GTWAY passing a chunk size larger than the SDSRV default chunk size. Verify the search results are returned using the SDSRV default chunk size.	TBD
EN_6B_03	20	1888	FC	Using the V0 EDG client or a test driver simulating V0 protocols, perform an inventory search. Set up the test case such that the inventory search results contain a fairly small amount of data, that fits within a single data chunk. Verify correct search results are returned to the V0 EDG client or the test driver. Verify the search results are returned in a single chunk from the SDSRV to the V0 GTWAY.	TBD
EN_6B_03	40	1890	FC	Using a test driver simulating SIPS, perform an inventory search through the MTMGW. Verify the correct search results are returned. Verify the search results are returned from the SDSRV to the MTMGW in one or more data chunks using the default chunk size configured in the SDSRV.	TBD
EN_6B_03	50	1891	FC	Perform multiple concurrent inventory searches from the following clients: 1. one or more V0 EDG client sessions or test drivers simulating V0 protocols 2. a test driver simulating the SIPS client. Verify the search results are returned to each V0 EDG client or test driver using proper data chunking. Verify the search results are returned from the SDSRV to the MTMGW using the default chunk size configured in the SDSRV.	TBD
EN_6B_03	60	1892	FC	Using the V0 EDG client or a test driver simulating V0 protocols, perform an inventory search specifying that only a subset of the metadata attributes are to be returned in the search results. Design the test such that the search results contain more granules than the SDSRV default chunk size. Also, let V0 GTWAY pass to the SDSRV a chunk size smaller than the SDSRV default chunk size. Verify correct search results are returned to the V0 EDG client or the test driver. Verify the search results are returned in multiple chunks from the SDSRV to the V0 GTWAY using the chunk size specified by the V0 GTWAY. Repeat the above inventory search test with the V0 GTWAY passing a chunk size larger than the SDSRV default chunk size. Verify the search results are returned using the SDSRV default chunk size.	TBD
EN_6B_03	70	1893	EC	Using the V0 EDG client or a test driver simulating V0 protocols, perform an inventory search. Design the test such that the SDSRV fails during the inventory results set chunking. Set up the test case such that the inventory search results contain more granules than the SDSRV default chunk size. Shutdown the V0 GTWAY connection to the SDSRV after the first chunk is returned from the SDSRV. Verify that the V0 GTWAY returns an error message to the V0 EDG client (or test driver) using V0 protocols.	TBD

Test Information			
Test Input	• Multiple data searches using the EDG Client and the SIPS Interface Test Driver.		
Test Output	• Search results specifying all metadata attributes at SDSRV default • Search results that exceed SDSRV default returned at SDSRV default • Search results that return V0 Gateway parameters for chunking less than SDSRV default • A successful search of less than one chunk that returns as a single chunk • A search using the SIPS driver that exceeds one chunk returned at SDSRV default • Multiple concurrent searches using EDG Client and SIPS driver returned at SDSRV default • Search containing a subset of metadata attributes that exceed SDSRV defaults returned ad SDSRV defaults • Search containing a subset of metadata attributes that are returned at V0 Gateway defaults • Search interrupted by a SDSRV failure that recovers • V0 Gateway connection failure that returns an error message to EDG Client		
Data Set Name		**Path Name**	**Description**
TBD		TBD	TBD
Test Configuration	Standard lab configuration, all servers running and available.		

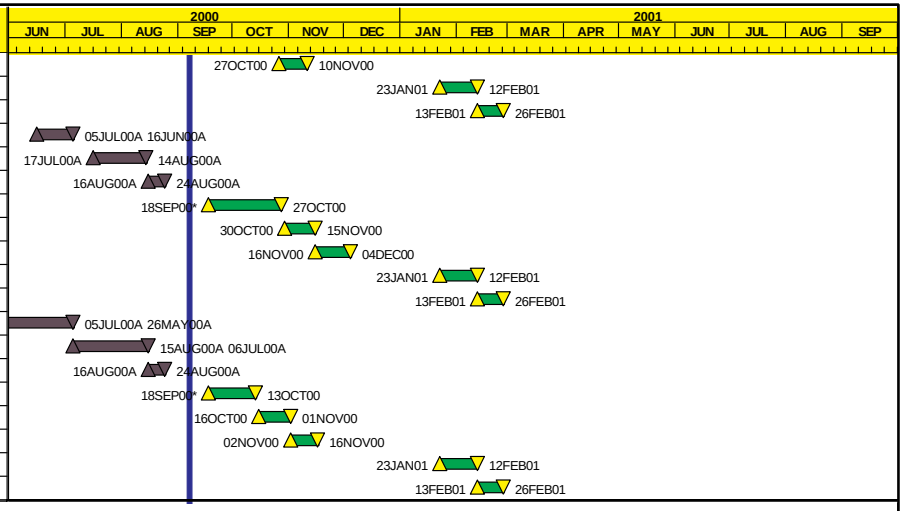
This page intentionally left blank.

Appendix B. Primavera Schedule Listing

The Primavera (P3) Schedule for the Release 6B Test Program is attached


Activity ID
Activity Description
2000
2001
JUN JUL AUG SEP OCT NOV DEC JAN FEB MAR APR MAY JUN JUL AUG SEP
6B Drop 6B
SVMGT6B010 Develop Test Plan Draft (DID 409) 15AUG00A 15DEC99A
SVACT6B020 Review 6B Tickets 01AUG00A 26MAY00A
SVACT6B030 Planning of 6B Test Procedures 15AUG00A 06JUL00A
SVACT6B050 Update Test Summaries 16AUG00A 24AUG00A
SVMGT6B040 Update Test Plan to Final (DID 409) 16AUG00A 21SEP00
SVACT6B060 Develop Draft 6B Test Procedures 15SEP00 27OCT00
SVACT6B070 Procedure Internal Review 10OCT00 15NOV00
SVACT6B080 ESDIS Test Procedure Review & Approval 27OCT00 04DEC00
SVACT6B100 VATC Release 6B Install & Checkout (2 modes) 16JAN01 29JAN01
SVMGT6B440 6B TRR (B/L at 1/22/01) 22JAN01*
SVACT6B140 Release 6B Dry Runs 23JAN01 22MAY01
SVACT6B150 6B Formal Acceptance Testing 08FEB01 26FEB01
SVMGT6B200 Test Ready for 6B CSR 26FEB01
SVACT6B160 6B Test Support to Transition Activities 02APR01* 11JUN01
SVACT6B190 DID 412 Test Report (30 days after final SRA) 22JUN01* 03AUG01
SVMGT6B900 Drop 6B Complete (SV tie off) 03AUG01
SVACT6B220 Review RM6B02 Enhanced Fault Rec-failovr Ticket 13NOV00 17NOV00
SVACT6B260 6B08010 - Enhanced Fault Rec-failover - Plan 06DEC00 19DEC00
SVACT6B290 6B08010 - Enhncd Fault Rec-failovr - Update Sum 18JAN01 14FEB01
SVACT6B330 6B08010 - Enhncd Fault Rec-failovr - Dev Draft 15FEB01 28MAR01
SVACT6B335 6B08010- Enhncd Fault Rec-failovr - Intrnl Rev 29MAR01 16APR01
SVACT6B340 6B08010 - Enhncd Fault Rec-failovr - ESDIS Rev 17APR01 01MAY01
SVACT6B345 6B08010 - Enhncd Fault Rec-failovr - Dry Run 02MAY01 22MAY01
SVACT6B350 6B08010- Enhncd Fault Rec-failovr - Formal 23MAY01 06JUN01
SVACT6B232 Review RH6B02 6B Data Types Ingest Ticket 27JUL00A 01AUG00A
SVACT6B272 6B09020 - 6B Data Types Ingest - Plan 02AUG00A 07AUG00A
SVACT6B302 6B09020- 6B Data Types Ingest - Update Summary 16AUG00A 24AUG00A
SVACT6B380 6B09020 - 6B Data Types Ingest - Dev Draft 18SEP00* 10OCT00
SVACT6B385 6B09020- 6B Data Types Ingest - Internal Rev 11OCT00 27OCT00
SVACT6B390 6B09020- 6B Data Types Ingest - ESDIS Rev 30OCT00 13NOV00
SVACT6B395 6B09020- 6B Data Types Ingest - Dry Run 23JAN01 07FEB01
SVACT6B400 6B09020 - 6B Data Types Ingest - Formal 08FEB01 13FEB01
SVACT6B202 Review EN6B01 SDSRV Failed Aquire Ticket 05JUL00A 16JUN00A
SVACT6B236 6B10020- SDSRV Failed Aquire - Plan 20JUL00A 15AUG00A
SVACT6B276 6B10020 - SDSRV Failed Aquire - Update Summary 16AUG00A 24AUG00A
SVACT6B405 6B10020- SDSRV Failed Aquire - Dev Draft 15SEP00 09OCT00
SVACT6B410 6B10020- SDSRV Failed Aquire - Internal Rev 10OCT00 26OCT00
Start Date 01SEP97
Finish Date 03AUG01
Data Date 08SEP00
Run Date 08SEP00 08:54
© Primavera Systems, Inc.
Early Bar
Progress Bar
Critical Activity
ECSP - SVER
Sheet 1 of 2
ECS Project
6B Test Schedule

Activity ID	Activity Description
SVACT6B415	6B10020- SDSRV Failed Aquire - ESDIS Rev
SVACT6B420	6B10020- SDSRV Failed Aquire - Dry Run
SVACT6B425	6B10020- SDSRV Failed Aquire - Formal
SVACT6B204	Review EN6B02 Request Priority Ticket
SVACT6B238	6B10030- Request Priority - Plan
SVACT6B278	6B10030- Request Priority - Update Summary
SVACT6B430	6B10030 - Request Priority - Dev Draft
SVACT6B435	6B10030- Request Priority - Internal Rev
SVACT6B440	6B10030 - Request Priority - ESDIS Rev
SVACT6B445	6B10030 - Request Priority - Dry Run
SVACT6B450	6B10030- Request Priority - Formal
SVACT6B206	Review EN6B03 Results Chunking Ticket
SVACT6B241	6B10040 - Results Chunking - Plan
SVACT6B282	6B10040 - Results Chunking - Update Summary
SVACT6B455	6B10040- Results Chunking - Dev Draft
SVACT6B460	6B10040 - Results Chunking - Internal Rev
SVACT6B465	6B10040- Results Chunking - ESDIS Rev
SVACT6B470	6B10040 - Results Chunking - Dry Run
SVACT6B475	6B10040 - Results Chunking - Formal



Start Date	01SEP97
Finish Date	03AUG01
Data Date	08SEP00
Run Date	08SEP00 08:54

Early Bar
Progress Bar
Critical Activity

ECSP - SVER **Sheet 2 of 2**

ECS Project

6B Test Schedule

This page intentionally left blank.